

IN THE CLAIMS:

1. (Original) An apparatus for managing power in a computer system, the apparatus comprising:

an operation system configured to set up a power mode of the computer system, wherein the power mode includes at least one of an operating mode and a power down mode;

at least one device configured to perform specific functions and operations;

at least one device driver configured to control operations on a corresponding device, wherein the device driver is configured to change the power mode of the corresponding device among the at least one of the operating mode and the power down mode; and

a filter driver coupled to the operation system, wherein the filter driver is configured to individually control a selected device to operate in the power down mode when the computer system is in the operating mode.

2. (Original) The apparatus according to claim 1, wherein the selected device is operated in the power down mode by the filter driver after receiving an operation signal at the selected device when the selected device is in the idle state longer than a predetermined time or the idle state of the selected device is not reset longer than the predetermined time.

3. (Original) The apparatus according to claim 1, wherein the filter driver monitors transceived packets between the operation system and the device driver and detects each device in the idle state.

4. (Original) The apparatus according to claim 1, wherein the filter driver generates and outputs a FIRP (False I/O request packet) configured to change the power mode of the corresponding device from the operating mode to the power down mode.

5. (Original) The apparatus according to claim 1, wherein the devices and the device driver comprise a sound card and a sound driver.

6. (Original) A method, comprising:
operating a computer system in first and second power modes;
operating devices in the computer system in the first and the second power modes; and
controlling a selected device in the second power mode when the computer system is in the first power mode.

7. (Original) The method according to claim 6, wherein the first power mode is an operating mode and the second power mode is a power down mode.

8. (Original) The method according to claim 7, wherein the controlling step comprises:

detecting at least one device that is in the idle state;

counting idle time for each of the detected devices in the idle state; and

changing the power mode of a corresponding device from the operating mode to a power down mode when the counted idle time for the corresponding device is greater than a prescribed amount.

9. (Original) The method according to claim 8, wherein the detecting and the counting are simultaneously performed.

10. (Original) The method according to claim 8, wherein the detecting step comprises:

monitoring transceived packets between an operation system in the computer system and device drivers; and

detecting the idle state of the devices.

11. (Original) The method according to claim 8, wherein the changing comprises:

generating a control message at the filter driver; and

transferring the control message to a corresponding device.

12. (Original) The method according to claim 8, wherein the changing step comprises:

generating a False I/O request packet (FIRP) similar to an I/O request packet of the computer system at the filter driver when the counted idle time is greater than the prescribed amount; and

transferring the FIRP to the corresponding device.

13. (Original) The method according to claim 8, wherein the changing step comprises operating the device in the operating mode if the accumulated idle time that has been counted is not greater than the prescribed amount.

14. (Original) The method of claim 8, wherein the prescribed amount has a first timeout value in a battery mode, a second timeout value in a performance mode, and the prescribed amount varies according to an object device, and wherein the prescribed amount or said at least one device is set by a user or preset.

15. (Original) The method according to claim 6, wherein the controlling comprising:
generating at an operation system, a power control message corresponding to the system power mode and outputting the power control message to a corresponding device;
changing or determining a power state of the corresponding device to set up

desired power states according to a kind of the power control message; and

dispatching the computer system based on a new power status.

16. (Original) The method according to claim 6, wherein the controlling step comprises independently controlling two of a plurality of devices or each of the devices to operate in the second mode when the computer system is in the first mode.

17. (Original) A method for managing power in a computer system, the method comprising:

detecting at least one device that is in the idle state when a power mode of the computer system is in an operating mode;

determining idle time of detected devices in the idle state; and

changing a power mode of a corresponding device from the operating mode to a power down mode when the idle state is not reset for a predetermined time.

18. (Original) The method according to claim 17, wherein a power manager provides a timer for checking respective devices in the idle state and managing power, and a filter driver changes a state of the devices after the predetermined time lapses.

19. (Original) The method according to claim 18, wherein the predetermined time has a first timeout value in a battery mode, a second timeout value in a performance mode, and the predetermined time varies according to an object device, and wherein the predetermined time or said at least one device is set by a user or preset.

20. (Original) The method according to claim 18, wherein when a device is operated a corresponding timer is reinitialized using the power manager.

21. (Original) A method for managing power in a computer system, comprising:
 setting up a power state of a corresponding device to a power down state;
 transferring a power control message for changing the corresponding device to an operating state to a device driver before transferring a message received for the corresponding device to the device driver;
 transferring the received message to the device driver after the corresponding device is changed to the operating state; and
 setting the corresponding device to the power down state after the device driver completes handling the message.

22. (Original) The method according to claim 21, wherein, if messages are received

continuously, the corresponding device is powered up at a first message and powered down at a last message.

23. (Original) The method according to claim 22, wherein the corresponding device is powered up when a first I/O request packet is taken out of a queue and powered down when the queue is empty or clear.

24. (Original) The method according to claim 23, comprising:
if an IRP is received from an IO manager in a Kernel mode, dispatching a routine by dispatch routines and receiving all kinds of packets; and
checking whether the received IRP is a power IRP, and if the received IRP is not a power IRP but an internal variable Suspend flag=0, placing a packet in the queue for normal filter driver operations.

25. (Original) The method according to claim 21, comprising:
if a packet enters into a queue, automatically dispatching an associated first device;
powering up the first device if the first device has been in the power down mode;
transferring one packet from the queue to an associated next device driver; and
completing a routine if the queue is empty, and if the routine is not dispatched again until a first timer is timeout, transferring at power manager, a power down IRP to the first

device to change the first device to the power down mode.

26. (Original) The method according to claim 25, wherein, if the first device is being used, resetting the first timer to prevent the first device from being in the power down mode.

27. (Original) The method according to claim 25, wherein the packet is given a lower priority in an operation system and thus dispatched after packets with higher priorities are first treated by the operation system, making the queue loaded with a plurality of packet IRPs.

28. (Original) The method according to claim 27, further comprising:

- if a received packet is a power IRP, checking whether the received packet is a system power IRP for changing the power state of the computer system;
- if the received packet is the system power IRP, checking whether the received packet is S0 being a new state among IRP data;
- if the received packet is not S0, setting an interval variable to Suspend flag = 1 since the computer system is now entering in the power down mode, and blocking any additional IRP to enter to the queue; and
- clearing all IRPs currently remaining in the queue.

29. (Original) The method according to claim 25, further comprising:

if a received packet is a power IRP, checking whether the received packet is a system power IRP for changing a power state of the system; and

if the received packet is the system power IRP, proceeding to a next device driver.

30. (Currently Amended) The method according to claim ~~30~~ 29, further comprising:

if the received packet is the power IRP, checking whether the received packet is the system power IRP for changing the power state of the system;

if the received packet is the system power IRP, checking whether the received packet is S0 being a new state among IRP data; and

setting an internal variable to Suspend flag=0 since the system is being enabled if the received packet is S0.

31. (Original) A method for managing power in a computer system, the method comprising:

detecting whether a device is in the idle state when a power mode of the computer system is in operating mode;

counting idle time for the detected device in the idle state; and

changing the power mode of the device from the operating mode to the power down mode when the idle time that has been counted is greater than a predetermined time.

32. (Original) The method according to claim 31, wherein the device is a sound card.

33. (Original) The method according to claim 32, wherein the predetermined time has a first timeout value in a battery mode, a second timeout value in a performance mode, wherein the predetermined time varies according to the device, and wherein the predetermined time or the device is set by a user or preset.